

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A) 平4-155533

⑤ Int.Cl.<sup>5</sup>  
G 06 F 9/46

識別記号 庁内整理番号  
3 5 0 8120-5B  
3 1 3 Z 8120-5B

④ 公開 平成4年(1992)5月28日

審査請求 未請求 請求項の数 1 (全9頁)

⑭ 発明の名称 仮想計算機制御方式

⑯ 特 願 平2-281371

⑰ 出 願 平2(1990)10月19日

⑱ 発 明 者 内 海 祐 之 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内

⑲ 発 明 者 井 上 愛 一 郎 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内

⑳ 出 願 人 富 士 通 株 式 会 社 神奈川県川崎市中原区上小田中1015番地

㉑ 代 理 人 弁 理 士 長 谷 川 文 廣 外2名

明 細 書

1. 発明の名称

仮想計算機制御方式

2. 特許請求の範囲

中央処理装置(1)と、主記憶(4)とを持つ実計算機上で複数の仮想計算機を実現する方式において、

仮想計算機および仮想計算機制御プログラムが保持している制御情報等を退避するための高速アクセス可能な退避メモリ(5)が備えられ、

退避メモリ(5)は仮想計算機番号により指定され、

退避メモリ(5)に仮想計算機および仮想計算機制御プログラムが保持している制御情報等を退避すると共に、退避メモリ(5)より復元することにより仮想計算機における状態遷移を行うことを特徴とする仮想計算機制御方式。

3. 発明の詳細な説明

[概要]

本発明は、実計算機上に複数の仮想計算機を実現する方式に関し、

低速な主記憶へのアクセスをすることなく高速に仮想計算機資源を退避、復元することを目的とし、

中央処理装置と、主記憶とを持つ実計算機上で複数の仮想計算機を実現する方式において、仮想計算機および仮想計算機制御プログラムが保持している制御情報等を退避するための高速アクセス可能な退避メモリが備えられ、退避メモリは仮想計算機番号により指定され、退避メモリに仮想計算機および仮想計算機制御プログラムが保持している制御情報等を退避すると共に、退避メモリより復元することにより仮想計算機間における状態遷移を行うことを特徴とする仮想計算機制御方式。

[産業上の利用分野]

本発明は、実計算機上に複数の仮想計算機を実現する方式に関する。

## 特開平4-155533(2)

実計算機上に複数の仮想計算機を実現するためには、実計算機上に二つの状態が設定される。

一つは、仮想計算機が動作する状態でVM状態と呼ばれる。他方の状態は仮想計算機を制御する状態でHYPER状態と呼ばれる。VM状態で発行された命令や割り込みがVM状態で処理できない場合には、HYPER状態へと遷移が移行し仮想計算機制御プログラムによりシミュレートが行われる。HYPER状態における仮想計算機制御プログラムが処理を終了すると、VM状態状態へと遷移しVM状態での処理が続けられる。

従来の仮想計算機制御方式では、仮想計算機あるいは仮想計算機制御プログラムの各種レジスタ等の制御情報(以下、資源と呼ぶ)は、主記憶に退避するようにしていた。

例えば、VM状態状態からHYPER状態への遷移の場合には、現在動作中の仮想計算機の資源を、主記憶の特定の領域へ退避し、主記憶の特定の領域に格納されている仮想計算機制御プログラムの資源をCPUへ復元し、HYPER状態を設

定するようにしていた。

そのため、従来の仮想計算機制御方式では、状態遷移における資源の退避、復元の度に主記憶へのアクセスを必要とし、オーバーヘッドの大きい非能率的なものであった。

本発明は、低速な主記憶へのアクセスをすることなく高速に仮想計算機資源を退避、復元することを目的とする。

(従来の技術)

第6図～第8図により従来の仮想計算機制御方式を説明する。

第6図は、仮想計算機における状態遷移図である。

61はVM状態、62はHYPER状態を示す。

図示の丸付の番号に従って説明する。

① VM状態61において処理不可能な命令や割り込みもしくはタイムアウト等が発生すると、そのことが仮想計算機制御プログラム(図示せず)に報告される。

② 仮想計算機プログラムは、仮想計算機資源の退避を開始し、終了するとHYPER資源の復元がなされる。

③ HYPER状態において、仮想計算機制御プログラムは制御がわたされた原因となった処理を認識し、それに対応した処理を開始する。

④ HYPER状態における必要な処理が終了する。

⑤ HYPER状態の退避を行う。

⑥ VM状態が復元され、仮想計算機上での処理が続行される。

第7図は、従来の仮想計算機制御方式を示す。

図(a)において、71は中央処理装置(CPU)、72はバッファ記憶であって、キャッシュメモリよりなるものである。73は主記憶(MSU)であって、仮想計算機制御プログラム(HYPER)退避領域と仮想計算機(VM)退避領域を含むものである。

73'はHYPER退避領域、73'-1~73'-nは、仮想計算機1(VM1)退避領域、

仮想計算機2(VM2)退避領域、仮想計算機n(VMn)退避領域である。

図(a)は仮想計算機制御プログラム(HYPER)とn個の仮想計算機(VM1~VMn)が動作しているシステムを示す。

仮想計算機1が退避する場合について、図(a)の動作説明をする。

VM状態からHYPER状態への遷移が必要になると仮想計算機1は仮想計算機制御プログラムへ報告を行い、仮想計算機制御プログラムが仮想計算機1の資源の退避を開始する。(仮想計算機が退避前に使用していた制御情報が格納されている)。

まず、CPU71内のレジスタ等の内容が読み出され、主記憶へ(73)格納するため、バッファ記憶(72)がアクセスされる。

バッファ記憶(72)がヒットすると、バッファ記憶(72)への書き込みが行われ、さらに、主記憶記憶(73)への書き戻しが行われる。バッファ記憶(72)がミスヒットの場合には退避

## 特開平4-155533(3)

領域の主記憶(73)の内容がブロックフェッチされて、バッファ記憶(72)へ格納される。そこで、再度バッファ記憶(72)がアクセスされてバッファ記憶(72)に退避データが書き込まれ、主記憶(73)におけるVM退避領域73'-1に書き戻され、仮想計算機Iの退避が終了する。

主記憶(73)への仮想計算機Iの退避が終了すると、HYPER退避領域からの資源の復元が開始される。

HYPER退避領域73'に格納されている内容を読み出すために、まず、バッファ記憶(72)がアクセスされる。バッファ記憶(72)がヒットするとバッファ記憶(72)に退避されていた値が読み出され、対応するレジスタへ書き込まれる。バッファ記憶がミスヒットした場合には主記憶へブロックフェッチ要求がだされHYPER退避領域73'の内容がバッファ記憶72へ格納され、再度バッファ記憶72がアクセスされ、必要な値が読み出されて復元される。

憶制御ユニット(図示せず)に出力し、仮想計算機番号によって選択されたバッファ記憶(76-1~76-n)の中で一致したときのみバッファ記憶がヒットしたとみなす。

このような構成において、VM状態におけるHYPER用バッファの書換えは行われていないときで、VM状態からHYPER状態への状態遷移においてはHYPER用のバッファ記憶(76-1~76-n)の中には前回使用した資源が必ず保存されている。そのため、HYPER状態の復元においては、ミスヒットすることなく高速に行える。しかし、これに先立って行われるVMの資源の退避においては、VM走行中にバッファ記憶(76-1~76-n)の置き換えが行われることがあるために、バッファ記憶が常にヒットするとは限らず、主記憶へのアクセスが必要になる場合を生じる。

第8図に従来の仮想計算機制御における退避、復元のフローを示す。

図において、80はVM状態、80'はHYPER

HYPER退避領域73'に退避されていた資源の復元が完了すると仮想計算機制御プログラムは制御がわたされた原因となった処理を認識し、それに対応した処理を開始する。

HYPER状態において必要な処理が終了すると、前述の処理と同様な処理により、HYPER状態からVM状態へと遷移し、仮想計算機Iの処理が続行される。

第7図(b)は、仮想計算機ごとにバッファ記憶を分割した場合を示す。

図において、75-1~75-n、77はセクタ(SEL)である。

76-1~76-nは各仮想計算機対応に分割したバッファ記憶である。

中央処理装置(CPU)74、主記憶(MS)78は図(a)におけるものと同じである。

図において、CPUは主記憶78のアクセスに際して、論理アドレスを動的アドレス機構により物理アドレスに変換し、物理アドレスによりアクセスする。同時に、仮想計算機番号をバッファ記

ER状態である。

図示の番号に従ってフローを示す。

81 VM状態における退避が開始され、全資源が退避されたか判断する。退避が全て終了したら86へ進む。全ての退避が終了していなければ、82へ進む。

82 バッファ記憶へアクセスする。

83 バッファアクセスがヒットしたか判断する。ヒットすれば85へ進む。ヒットしなければ84へ進む。

84 主記憶から退避領域のブロックフェッチをしてバッファ記憶に格納する。そこで、再度82のバッファ記憶アクセスを繰り返す。

85 83においてヒットすればVM資源を主記憶に格納し、81以降の処理を繰り返す。

86 81でVM状態における全資源退避が終了すると、HYPER資源の復元を開始する。

HYPER資源の格納が全て終了すると、HYPER状態における処理が開始される。

全資源の復元が終了していなければ、87へ進

む。

87 バッファ記憶へアクセスする。

88 ヒットすれば、90へ進む。ヒットしなければ89へ進む。

89 主記憶からHYPER領域のブロックフェッチを行い、バッファ記憶へ格納する。そこで再度、バッファ記憶アクセスを行う。

90 88でヒットすれば、HYPER資源を主記憶へ格納し、86以降の処理を繰り返す。

〔発明が解決しようとする課題〕

従来の技術においては、状態遷移における退避、復元の度に、主記憶へのアクセスを必要とし、オーバーヘッドの大きいものであった。

本発明は、状態遷移において、主記憶にアクセスすることなく仮想計算機資源を退避、復元することにより、状態遷移に必要な時間を短縮することを目的とする。

〔課題を解決するための手段〕

M状態で処理できない命令および割込みが認識されると、仮想計算機Iは仮想計算機制御プログラムに報告する。報告を受けた仮想計算機制御プログラムは仮想計算機Iの資源であって中央処理装置1におけるPSW(Program Status Word), EG(General Register), AR(Access Register), FPR(Floating Point Register) CR(Control Register), PR(Prefix Register), CK(Time-of-day Clock), CKC(Clock Comparator), PT(CPU Timer)の内容を読み出す。そして、退避メモリ5におけるVM1退避領域に格納する。

退避が終了すると、仮想計算機制御プログラムの動作環境を設定するために、退避メモリ5のHYPER領域に退避していた資源を中央処理装置1へ読み込み、対応するレジスタ等へ復元する。

復元が終了すると、仮想計算機制御プログラムは制御が渡された原因となった処理を認識し、それに対応した処理を開始する。

必要な処理が終了すると、制御プログラムは自分が使用していた資源を退避メモリ5のHYPER

#### 特開平4-155533(4)

本発明は、物理アドレスによってアドレス指定されるバッファ記憶および主記憶とは独立に、仮想計算機番号によってアドレス指定される高速アクセス可能なメモリを実計算機上に設け、仮想計算機および仮想計算機制御プログラムの資源の退避、復元を行うようにした。

第1図に本発明の基本構成を示す。

図において、1は中央処理装置(CPU)、2は動的アドレス変換機構(DAT)であって、論理アドレスを物理アドレスに変換するものである。3はバッファ記憶であって、キャッシュメモリ(CACHE)よりなるもの、4は主記憶(MSU)、5は退避メモリであって、仮想計算機(VM1~VMn)および仮想計算機制御プログラム(HYPER)の退避領域、6は仮想計算機番号及び仮想計算機資源オフセットである。

〔作用〕

第1図の基本構成の動作を説明する。

例えば、仮想計算機I(VM1)の動作中にV

R退避領域に書き込み、仮想計算機Iの資源を退避メモリ5のVM1退避領域から読み出し、中央処理装置1のレジスタ等への復元を行い、仮想計算機Iへ制御を返す。

退避メモリ5における各資源の退避および復元には仮想計算機番号、仮想計算機資源オフセット6によりアドレスを指定する。

なお、第1図に構成においては、動的アドレス変換機構(DAT)2が備えられているが、主記憶主記憶(MSU)4へのアクセスは論理アドレスにより行う場合もある。

第2図により、本発明における退避、復元のフローを示す。

図において、20はVM状態、20'はHYPER状態である。

VM状態からHYPER状態へ遷移する場合について、図示の番号に従ってフローを説明する。

21 全VM資源の退避が終了したか判断する。終了していれば23へ進み、終了していなければ22へ進む。

22 VM資源をVM退避領域に格納し、21以降の処理を繰り返す。

23 全VM資源の退避が終了すると、HYPER資源の復元を開始する。そして、全HYPER状態資源が復元されるとHYPER状態が設定される。復元が終了していない場合には、24に進む。

24 HYPER資源をHYPER格納領域に格納し、23以降の処理を繰り返す。

第3図は、本発明における退避メモリへのアドレス計算方法を示す。

図において、31は仮想計算機番号、32は仮想計算機資源オフセット、33は仮想計算機番号と仮想計算機資源オフセットを加算して得られる仮想計算機資源格納アドレスである。

34は退避メモリ、35は退避メモリ34のアドレスであって、仮想計算機番号0、1～Nと仮想計算機資源(PSW0、PSW1、GR0等)のオフセットにより定められるものである。

図は、仮想計算機番号nにおける各仮想計算機

各種レジスタ備えるものである。46は退避メモリ、49は資源アドレスを設定した領域であって、仮想計算機資源であるレジスタのオフセット値をもつ領域、47は読み出しバス、48は書き込みバスである。

仮想計算機の資源であるレジスタ類はレジスタック(RGSTK)45に備えられている。

退避メモリ46の各HYPER領域、VM領域は、レジスタック(RGSTK)45への書き込みのために書き込みバス48を介して送られてくるデータを同時に受けとり、それぞれの領域の常に更新された新しい値を保持している。

また、仮想計算機の資源を復元する場合には、読み出しバス47を通じて退避メモリ46に保存されているデータを読み出し、書き込みバス48を介して、レジスタック(RGSTK)45へ書き込み、VM状態もしくはHYPER状態を復元する。

退避メモリ46へアクセスするアドレスは仮想計算機番号と仮想計算機資源オフセット(資源領

## 特開平4-155533(5)

資源(PSW1、PSW2等)のオフセットにより定められたそれぞれの資源のアドレスを示す。

本発明においては、主記憶4には仮想計算機および仮想計算機制御プログラムの資源の退避領域は設定されない。そのため、仮想計算機システムにおいて、VM状態とHYPER状態の遷移における退避、復元が主記憶4にアクセスされることなく、高速な退避メモリにのみアクセスすればよいので、状態遷移を高速に行うことができる。

### 〔実施例1〕

第4図は、本発明における退避メモリをCPUに構成した場合の実施例を示す。

図において、40はCPU、41はCPU40におけるI-ユニットであって、各種レジスタ類を備える部分である。42はE-ユニットであって、演算領域である。43はS-ユニットであって、キャッシュメモリ等のバッファ記憶領域であるもの、44はMSU(主記憶)である。

45はレジスタック(RGSTK)であって、

域アドレス49)により生成される。

### 〔実施例2〕

第5図により本発明の実施例2を説明する。

図は、バッファ記憶(キャッシュメモリ)の領域の一部を退避メモリ専用に固定して使用する場合を示す。

通常バッファ記憶は、主記憶(MSU)の異なる複数の領域のコピーを保持するためにウェイと呼ばれるブロックに分割されている。本実施例は、この複数のウェイの中の1つを退避メモリ専用に用いるようにした。

図において、50はCPU、51はI-ユニット、52はE-ユニット、53はS-ユニットであって、バッファ記憶(キャッシュメモリ)を備えるもの、54はMSU(主記憶)である。

55は資源アドレスを保持している領域、56は論理アドレスを保持している領域、57は動的アドレス変換機構(DAT)である。

59-1～59-nはバッファ記憶であって、

n個のウェイ0、ウェイ1、……ウェイnに分割されているものである。

本発明においては、例としてウェイ0（59-1）を退避メモリ専用を使用する。

ウェイ0は他のウェイと異なり、MSU（主記憶）のコピーは保持しないため主記憶との接続は不要である。

状態遷移に際して、I-ユニットから資源の退避を示す信号とデータが送られてくる。この時、S-ユニットはこの信号を受取りI-ユニットへバッファヒットを示す信号を返し、ウェイ0を選択してデータを退避させる（バッファは必ずヒットする）。I-ユニットから次にI-ユニットから資源を復元させるための信号が送られてくると、S-ユニットはこの信号を受取りI-ユニットへバッファ記憶のヒットを示す信号を返し、ウェイ0をウェイにおいて選択要求されているデータをI-ユニットへ送る。

退避メモリとして使用するバッファ記憶（59-1）へのアクセスに必要なアドレスは仮想計算

る。

第6図は、仮想計算機における状態遷移図である。

第7図は、従来の仮想計算機制御方式を示す図である。

第8図は、従来の退避・復元のフローを示す図である。

第1図において、

- 1 : 中央処理装置、
- 2 : 動的アドレス変換機構（DAT）、
- 3 : バッファ記憶（キャッシュメモリ）、
- 4 : 主記憶（MSU）、
- 5 : 退避メモリ、
- 6 : 仮想計算機番号、仮想計算機オフセット。

特許出願人 富士通株式会社  
代理人弁理士 長谷川 文廣（外2名）

## 特開平4-155533(6)

機番号と仮想計算機資源オフセット（資源アドレス）により生成する。

〔発明の効果〕

本発明によれば、仮想計算機システムにおいてVM状態とHYPER状態の遷移における仮想計算機資源の退避、復元を、主記憶にアクセスすることなく行うことができる。そのため、状態遷移を高速に行うことができ、仮想計算機システムの処理が高効率化される。

### 4. 図面の簡単な説明

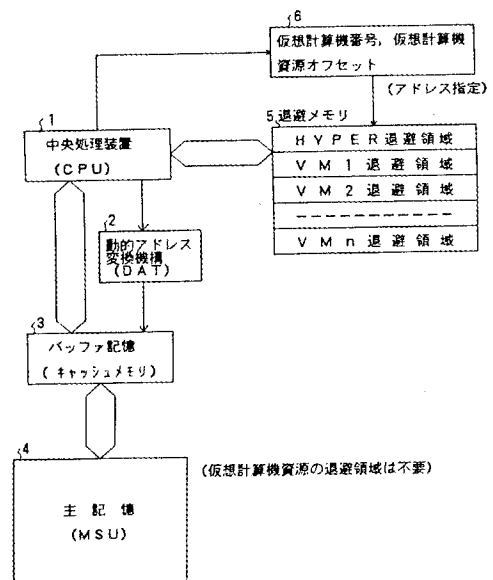
第1図は、本発明の基本構成を示す図である。

第2図は、本発明における退避・復元のフローを示す図である。

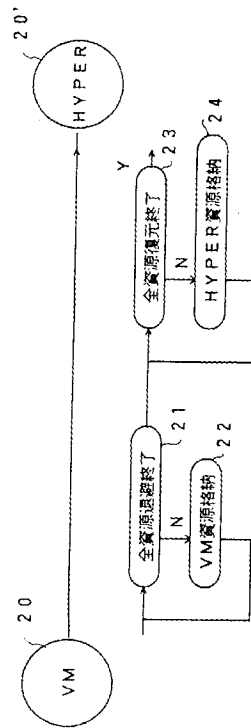
第3図は、本発明におけるアドレス計算方法を示す図である。

第4図は、本発明の実施例（1）を示す図である。

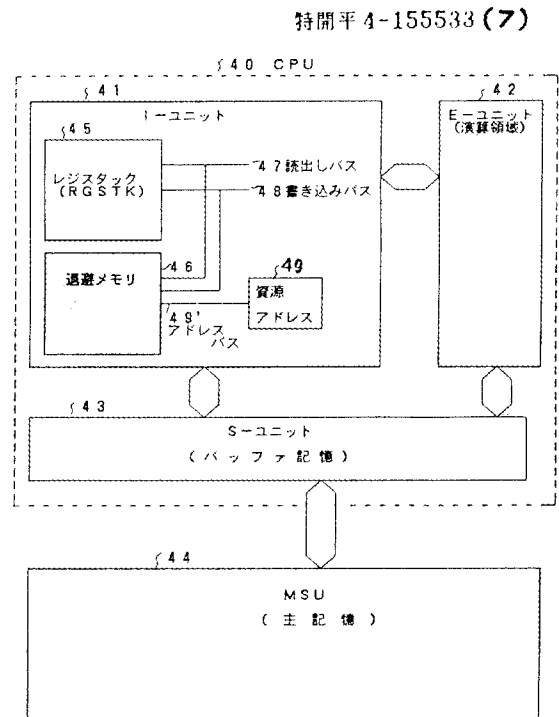
第5図は、本発明の実施例（2）を示す図である。



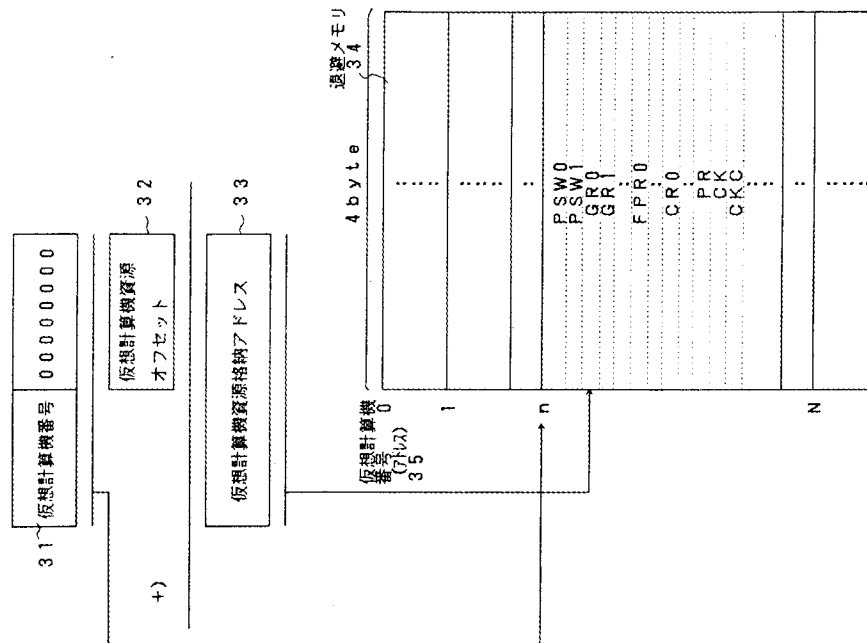
本発明の基本構成図  
第1図



本発明における回避・復元フロー  
第2図

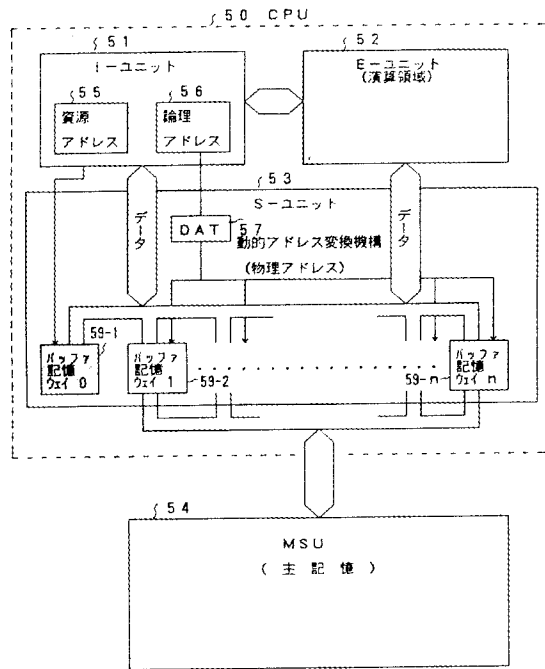


本発明の実施例(1)  
第4図

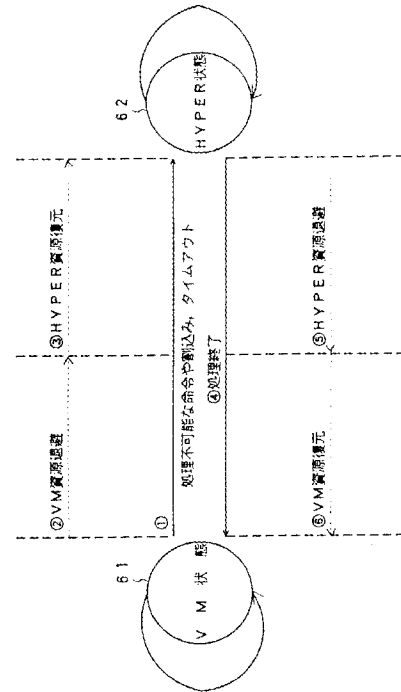


本発明におけるアドレス計算方法  
第3図

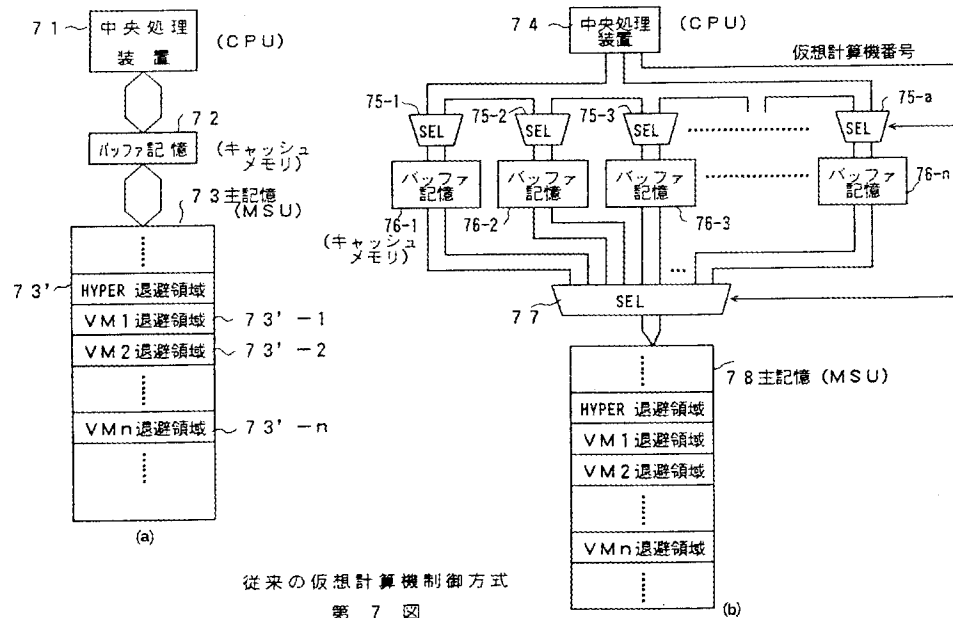
特開平4-155533 (8)



本発明の実施例(2)  
第5図



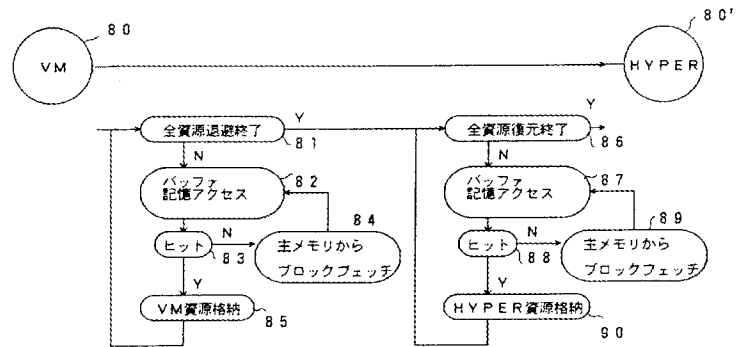
仮想計算機における状態遷移図  
第6図



従来の仮想計算機制御方式  
第7図



特開平4-155533 (9)



従来の退避・復元のフロー

第8図